Chapter 1. Using dCache to Copy Files to/from Enstore

Table of Contents

Whenever a client application needs to talk to the dCache, it has to choose an appropriate door into the system. For each door, there are corresponding utilities for copying files back and forth between your machine and your `/pnfs/storage-group` area on the machine running dCache. We describe how to use the supported utilities in this chapter.

Each dCache server may have multiple doors, thus allowing a variety of access methods. Each door is limited to about 50 simultaneous transfers; more doors can be added as needed. The dCache supports Kerberos V5 for FTP, the dCache native dCap C-API, and GSI FTP.

Caution

The dCache server node and the ports documented in this section are subject to change. You can always find the current configuration from the web page [http://www-isd.fnal.gov/enstore/dcache_user_guide.html].

dCache-Native dCap

dCap is a dCache-native access protocol. The dCap client, dccp, is available at the KITS ftp site [ftp://fnkits.fnal.gov/products/dcap/]. The libdcap library provides POSIX-like open, create, read, write and lseek functions to the dCache storage. In addition there are some specific functions for setting debug level, getting error messages, and binding the library to a network interface. See the dCache manual for usage information [http://www-dcache.desy.de/manuals/libdcap.html].

Authentication Mechanisms

There are three authentication mechanisms used for the dcap protocol:

1. Plain
2. Kerberos
3. X509

All three have separate port numbers and separate "setup dcap" qualifiers for the UPS/UPD distribution of dCap. CDF has both kerberized dcap and X.509 dcap.

These different qualifiers have to be setup correctly in UPS for this to work though, with a ups listing for each qualifier state. For debugging this issue, the env var DCACHE_IO_TUNNEL should point to the appropriate shared library for the authentication mechanism: a file like libgsstunnnel.so for krb5, libgsitunnel.so for x509, and it should be unset for plain dcap access.

Plain dCap

Plain dCap is strictly limited to fnal.gov domain access only. It uses uid/gid permissions on files in PNFS. Plain dcap is not the same as weakly-authenticated FTP, as it does allow write access as one's uid/gid permits. On FNDCA1, plain dcap is available on fndca1.fnal.gov:24125 and 24136. The UPS setup command reads:

```
$ setup dcap -q unsecured
```

Kerberized dCap

Note

If your dCap door uses Kerberos V5 authentication, you must have a Kerberos principal for the FNAL.GOV realm.

Kerberized dcap is available on ports 24725, 24736 to anyone with valid FNAL.GOV kerberos credentials. Install the dCap product on your computer. See the dCap Setup Manual [http://www-dcache.desy.de/manuals/dcap_setup.html].

The UPS setup command reads simply:

`setup dcap`

Besides creating a certificate with the "kx509", you have to place the certificate in the correct format and the correct location. Please see “Using Globus tools for submitting grid jobs from Linux/UNIX” [http://security.fnal.gov/pki/Get-Personal-DOEGrids-Cert.html#globus] for the current suggested means of doing this.

Nodes and Ports

The nodes and ports available for dCap are subject to change; to get a current listing, run the following command, using your storage group (sample output shown for storage group cdfen):

```
% cat '/pnfs/cdfen/.(config)(dCache)(dcache.conf)'
cdfdca1.fnal.gov:25125
cdfdca1.fnal.gov:25136
...
cdfdca2.fnal.gov:25153
cdfdca2.fnal.gov:25154
cdfdca3.fnal.gov:25155
...
```

The dCap protocol requires specification of the dCache server host, port number, and domain, in addition to the inclusion of “/usr” ahead of the storage group designation in the PNFS path. Its structure is shown here:

`dcap://serverHost:port/</pnfs>/storage_group/usr/filePath`

There are supposed to be two slashes inbetween the port number and pnfs (e.g., .. :24124//pnfs/...) but since users frequently just put one slash, we’ve allowed either one or two.

Note

If you run any of the following commands (**dccp**, **dc_check**, **dc_stage**) and it fails because the port is unavailable, try the command again with a different port number, or with a different host and port combination.

dccp

dccp, which is available in the dCap product, provides a cp-like functionality on the pnfs file system. It has the following syntax:

`dccp` [-d *`debuglevel`*] [-h *`relpyHostName`*] [-i] [-S]
[-P [-t *`time`*] [-l *`location`*]] [-b *`read-ahead bufferSize`*]
[-B *`bufferSize`*] [-u] [-w] [-p *`first-port`* [:last-port]]
[-T *`IO tunnel plugin`*] {*`source`*} [*`destination`*]

See the *dCache Manual* for more on options and command usage [http://www-dcache.desy.de/manuals/dccp.html].

dc_stage

dc_stage prestages the request; for read requests only. It is particularly useful when you’d like to grab the file quickly from the dCache when you’re ready for it. Use this with the `-t` option to set an interval of

time between the download to the dCache and the download from the dCache to your local system. If `-t` is not used, the default interval is zero.

`dc_stage` [-t *`number of seconds`* {*`source`*} [*`destination`*]]

dc_check

The **dc_check** command checks if a file is on disk in the `dCache.dc_check` file

PNFS Not Mounted Locally: Syntax and Examples

If PNFS is not mounted locally (the general case), you'll have to supply the protocol, node, port, and pnfs directory for the remote location (the "source" on reads, and the "destination" on writes).

Requesting a write to Enstore

```
% dccp path/to/local/file \
dcap://<serverHost>:<port>///pnfs/fnal.gov/usr/
<storage_group>/<filePath>
```

Requesting a write from your local /tmp directory

```
% dccp /tmp/myfile \
dcap://cdfdca1.fnal.gov:25140//pnfs/fnal.gov/usr/cdfen/x/myfile
```

Checking if a file is on disk in the dCache by running dc_check

```
% dc_check \
dcap://fndca1.fnal.gov:24725//pnfs/fnal.gov/myfile
```

For a read rather than a write:

```
% dccp \
dcap://cdfdca1.fnal.gov:25140//pnfs/fnal.gov/usr/cdfen/x/myfile \
/tmp/myfile
```

Pre-staging this request with an hour interval using dc_stage:

```
% dc_stage -t 3600 \
dcap://cdfdca1.fnal.gov:25140//pnfs/fnal.gov/usr/cdfen/x/myfile \
/tmp/myfile
```

pnfs Mounted Locally: Syntax and Examples

If pnfs is mounted on your local machine, you only need to specify the simple pnfs path of the remote file, e.g. (for a write):

Specifying the pnfs path of the remote file

```
% dccp path/to/local/file/pnfs/<storage_group>/<filePath>
```

Writing the file to Enstore

```
% dccp /tmp/myfile /pnfs/cdfen/x/myfile
```

Reading the file from Enstore

```
% dccp /pnfs/cdfen/x/myfile /tmp/myfile
```

Grid (GSI) FTP

GSI stands for Grid Security Interface. GSI FTP uses Grid Proxies for authentication and authorization and is compatible with popular Grid middleware tools such as **globus-url-copy** (available in the Globus toolkit at Globus [http://www.globus.org/] or **sam_gridftp** in Kits). The dCache GSI FTP currently runs on port 2811 on the following door nodes (different nodes for different user groups):

General users	fndca1
CDF	cdfdca1, cdfdca2, cdfdca3
CMS	cmsdca1, cmsdca2 and cmsdca3

It is more convenient to run this through an interface like srmcp which allows you to perform multiple transfers in a single command. In addition, it optimizes the parameters of the transfer, and allows FTP to scale with user load (overcoming a passive gridftp protocol issue).

Obtain Grid Proxies

Globus tools require that a user be authenticated with a short-term authentication Grid proxy. This proxy is created from (long-term) X.509 credentials issued by the DOE science grid [http://www.doegrids.org/] or other Certificate Authority [http://computing.fnal.gov/security/pki], or from Kerberos credentials at Fermilab. DOE science grid is the recommended source for an X.509 certificate. We recommend that you use the command grid-proxy-init to generate your proxy from your certificate. A proxy expires after a preset duration, and then a new one must be regenerated from the user's (long-term) X.509 certificate.

X.509 Grid proxies can be issued automatically for Fermilab users authenticated to Kerberos. (See Fermilab instructions [http://computing.fnal.gov/security/pki/]. This involves downloading a KX.509 certificate. KX.509 can be used in place of permanent, long-term certificates. It works by creating X.509 credentials (certificate and private key) using your existing Kerberos ticket. These credentials are then used to generate the Globus proxy certificate. More on KX.509 [http://www.ncsa.uiuc.edu/~aloftus/NMI/kx509.html].

GSI FTP with globus-url-copy

Install the Globus toolkit [http://www.globus.org/]. Run the **globus-url-copy** command in order to use the GSI FTP protocol to transfer files. Use the `gsiftp://` URL prefix for the PNFS (Enstore) path, and `file://` for the other URL.

For example, to copy from Enstore:

```
% globus-url-copy \
gsiftp://[[<src_node>:]port]/<source_url_path> \
file://[[<dest_node>]:port]/<dest_url_path>
```

To copy to Enstore:

```
% globus-url-copy \
file://[[<src_node>:]port]/<source_url_path> \
gsiftp://[[<dest_node>]:port]/<dest_url_path>
```

For a CDF user copying from Enstore to a local disk:

```
% globus-url-copy \
gsiftp://cdfdca1.fnal.gov:2811/<pnfs_path> \
file://<local_url_path>
```

Copying from one Enstore system to another (here, from CDFDCA to FNDCA):

```
% globus-url-copy
gsiftp://cdfdca1.fnal.gov:2811/<pnfs_path>\
gsiftp://fndca1.fnal.gov:2811/<pnfs_path>
```

Storage Resource Management (SRM)

SRM is middleware for managing storage resources on a grid. The SRM implementation within the dCache manages the dCache/Enstore system. It provides functions for file staging and pinning, transfer protocol negotiation and transfer url resolution.

Note

Pinning refers to making a file undeletable in the cache for the period of time called the “lifetime of the job”.

The SRM client srmcp provides a convenient way to transfer multiple files from/to Enstore via dCache using a variety of protocols. More on SRM... [http://grid.fnal.gov/]

Srmcp is the implementation of SRM client as specified by the SRM spec [http://sdm.lbl.gov/srm/documents/joint.docs/srm.v1.0.doc]. You can use srmcp for the retrieval and/or storage of files to/from Enstore (or other Mass Storage Systems which implement SRM, e.g., SLAC’s, CERN’s). In this document we focus on file transfers to/from Fermilab’s Enstore via dCache.

Preparing to Use srmcp

Two packages are available, one with java (srmcp), the other with a C-based client (srmtools); they are both in Kits [ftp://fnkits.fnal.gov:8021/products/]. To use the java-based srmcp, you will need to install java on your system. You will also need to install either the globus toolkit or dccp, depending on which protocol you wish to use. In order to use GSI with srmcp, follow the instructions in the README.SECURITY file that comes with srmcp in Kits.

Command Syntax

`srmcp` [options] {source(s)} {destination}

Default options will be read from a configuration file but can be overridden by command line options. The options are listed and defined in the srmcp README file in Kits. We do not list them here.

The SRM protocol, used for the remote file specification, requires the SRM server host, port number, and domain. For the fnal.gov domain, the inclusion of "/usr" ahead of the storage group designation in the PNFS path is also required.

srm://*serverHost*:*portNumber*/*root of filesystem*/*storage_group*[/usr]/*filepath*

Some examples, the first two for the fnal.gov domain, the third for cern.ch:

- **srm://cdfdca1.fnal.gov:8443//pnfs/fnal.gov/usr/cdfen/filesets/*filePath***
- **srm://fndca1.fnal.gov:8443//pnfs/fnal.gov/usr/*filePath***
- **srm://wacdr002d.cern.ch:9000/castor/cern.ch/user/*filePath***

Examples

These examples are taken from the srmcp v1_2 README file in Kits (with unnecessary options removed).

The following command will retrieve two files, `/mypath/myfile1.ext` and `/mypath/myfile2.ext`, from Enstore via dCache (for a CDF user) and store them in the user's local directory `/home/me/targetdir`.

Note that srmcp requires that the PNFS path include `/pnfs/fnal.gov/usr/` ahead of the storage group designation.

```
% srmcp \
srm://cdfdca1.fnal.gov:8443//pnfs/fnal.gov/usr/cdf/myfile1.ext \
srm://cdfdca1.fnal.gov:8443//pnfs/fnal.gov/usr/cdf/myfile2.ext \
file://localhost//home/me/targetdir
```

The following will copy the same files from one Enstore installation (CDFEN) to another (STKEN):

```
% srmcp \
srm://cdfdca1.fnal.gov:8443//pnfs/fnal.gov/usr/cdf/myfile1.ext \
srm://cdfdca1.fnal.gov:8443//pnfs/fnal.gov/usr/cdf/myfile2.ext \
srm:/fndca1.fnal.gov:8443/targetdir
```

The following will get the file using dccp client, overriding the default (dccp would have to be already installed on you machine):

```
% srmcp \
-protocols=dcap \
srm:/fndca1.fnal.gov:8443//pnfs/fnal.gov/usr/targetdir/myfile1.ext \
file:////tmp/myfile1.ext
```

Note

The four slashes in the last line refer to: `file://` ; host, which comes next, is " "; path is

/tmp/.... *[your_login_id@]*fndcal: *pnfs_path /path/to/local_file*

X.509 dCap

X.509 dcap is available on ports 24525, 24536. The UPS setup command reads:

```
$ setup dcap -q x509
```

For authentication to work, the environment variable X509_CERT_DIR must be set. If not, check with the compute administrator to get globus setup correctly for your job.

GSI FTP with kftpcp *(Deprecated)*

GSI FTP is also available with kftpcp (see the section called “Kerberized FTP via the kftpcp Command”). Install and setup kftp (from Kits [ftp://fnkits.fnal.gov:8021/products/kftp]). Also from kits, install and setup **gsspy_gsi** (for Grid proxy) instead of **gsspy_krb**. Kftpcp works the same as described in section 5.4 except that the port number is 2811 in this case.

We refer you to section 5.4 for details, but here’s a quick example for a general user (using STKEN) to copy from Enstore to a local disk:

% kftpcp -p 2811 -m p [-v]

Simple Kerberized FTP

The dCache door for Kerberized ftp service enforces Kerberos authentication (see the Strong Authentication at Fermilab Documentation [http://computing.fnal.gov/docs/strongauth/]). It currently runs on the following nodes and corresponding ports:

1. fndca1.fnal.gov, port 24127 (for STKEN)

2. cdfdca1, 2 and 3, port 25127 (for CDFEN)

The port number is installation-specific.

Any Kerberized ftp client can be used on the client machine. You must specify the host port in your ftp command.

Caution

File read and write functionality is supported when the user (a) is authorized by the experiment to access the data stores, and (b) has obtained Kerberos credentials.

Important

Portal Mode (CRYPTOCard) access is not supported since it is not compatible with automated transfers or future GRID development.

Prepare to use Kerberized FTP

In order to establish the kftp service on dCache, you must first:

- Have a valid Fermilab UNIX account (UID and GID)
- Have a Kerberos principal for FNAL.GOV (if Kerberized access is required)
- Ask your experiment’s Enstore liaison to register you for the service. You will need to provide the following information to the liaison:

1. username
2. UID and GID (run the command id at the UNIX prompt to find their values)
3. storage group
4. root path under `/pnfs/<storage_group>/...`
5. Kerberos principal(s), if applying for Kerberized door
6. password, if applying for weak door (request by emailing `<dcache-admin@fnal.gov>`)

Warning

This is for groups, not individuals.

- *Optional:* Install the kftp product from KITS. kftp is useful for running scripts to transfer files. To do so, run:

$ setup upd$ upd install -G "-c" kftp

Sample Kerberized FTP session

User is authenticated to Kerberos and authorized for the Kerberized dCache door (currently at fndca1.fnal.gov, port 24127):

```
% ftp fndca1.fnal.gov 24127

Connected to stkendca3a.fnal.gov.
220 FTPDoorIM+GSS ready
334 ADAT must follow
GSSAPI accepted as authentication type
GSSAPI authentication succeeded
Name (fndca:aheavey):
200 User aheavey logged in
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd aheavey/test3

250 CWD command succcessful. New CWD is
</aheavey/test3>
ftp> ls
200 PORT command successful
150 Opening ASCII data connection for file list
dupl2
duplexps
226 ASCII transfer complete
ftp> get duplexps
local: duplexps remote: duplexps
200 PORT command successful

150 Opening BINARY data connection for
/pnfs/fs/usr/test/aheavey/test3/duplexps
```

```
226 Closing data connection, transfer successful

42 bytes received in 0.033 seconds (1.2 Kbytes/s)

ftp>
```

Kerberized FTP via the kftpcp Command

In order to access data from a batch job or a background process, you should either use ftp client libraries (available from many sources), or the kftp package. This package includes a Kerberized client library and a GSI client library; you can use either. A regular ftp client (Kerberized or not) is an interactive program which is hard to use in batch mode.

To use the product in a UPS environment as a Kerberized FTP client, first run:

% setup gsspy_krb; setup kftp

Then run the kftpcp command to copy one or more files. This command can be used from the shell or in a script.

Syntax and Options

% kftpcp [<options>] *source_file destination_file*

The available options include:

-p *port*	ftp server port number
-m [a | p]	ftp server mode. Active (default), or passive
-v	verbose mode

Important

If your login id is the same on fndca1 and your local system, and if they match your Kerberos principal, you can leave off **\<*your_fndca1_login_id*\>@** in front of **fndca1:** in the command.

Important

Depending on how your access is configured, typically you only need to specify the path to the remote file starting from the directory under your /pnfs// area. For example, to specify the remote file /pnfs/my_storage_group/path/to/file on the command line, enter only **/path/to/file**, including the initial slash. You can use the full specification (starting with **/pnfs//usr/**)

Download a File

To download a stored data file from Enstore via the dCache, using fndca1 as a sample server host, run:

% **kftpcp -p 24127 -m p [-v]** ***login_id*@fndca1:*/path/to/remote_file /path/to/local_file>**

Upload a File

To upload a new data file, again using fndca1, run:

**kftpcp -p 24127 - m p [-v] *`/path/to/local_file`* [*`fndcal_login`*]@fndcal:*`/path/to/re`*

Examples

To read (download) the stored file `/pnfs/storage_group/mydir/myfile` into a local file of the same name, run:

```
% setup kftp
% kftpcp -p 24127 -m p -v myloginid@fndca1:/mydir/myfile \
/path/to/myfile
Transferred 42 bytes
```

Or, if your usernames and principal all match, you could shorten it to:

```
% kftpcp -p 24127 -m p -v fndca1:/mydir/myfile /path/to/myfile
```

Weakly-Authenticated FTP Service (Read-only)

The dCache weakly-authenticated ftp service currently runs on node the following nodes and corresponding ports:

1. fndca1.fnal.gov, port 24126 (for STKEN).

2. cdfdca1, 2, and 3, port 25126 (for CDFEN)

This is read-only, and is not necessarily allowed by all experiments. This ftp service can be accessed by ordinary ftp client software. You must specify the host port in your ftp command, as shown below. The Enstore admin will have sent you an email to confirm your registration for this service, and included a password for it. This is a weak password. Log in with your username and password.

Sample weakly-authenticated read-only ftp session

Here we explicitly use a weakly-authenticated ftp client, `/usr/bin/ftp`, and make the connection to fndca port 24126. In the session, we first successfully retrieve a file called myfile, and secondly attempt to write a file trace.txt and (correctly) fail.

```
% /usr/bin/ftp fndca1.fnal.gov 24126

Connected to stkendca3a.fnal.gov.
220 FTPDoorIM+PWD ready (read-only server)
Name (fndca:aheavey):
331 Password required for aheavey.
Password: XXXXXXXXXXXXXXX
230 User aheavey logged in
ftp> cd aheavey/test3

250 CWD command succcessful. New CWD is
</aheavey/test3>
```

```
ftp> ls
200 PORT command successful

150 Opening ASCII data connection for file list

myfile
myfile2
myfile3
226 ASCII transfer complete

10 bytes received in 0.018 seconds (0.55 Kbytes/s)

ftp> get myfile
200 PORT command successful
150 Opening BINARY data connection for
/pnfs/fs/usr/test/aheavey/test3/myfile

226 Closing data connection, transfer successful

local: myfile remote: myfile

42 bytes received in 0.05 seconds (0.82 Kbytes/s)

ftp> put trace.txt
200 PORT command successful
500 Command disabled
ftp> bye
```

Important

If you need to change this password, send email to <dcache-admin@fnal.gov>.